The interplay between numerical solvers and computational geometry for exascale computing

Ivan Yotov[1] and Paolo Zunino[2,*]
[1] Department of Mathematics, University of Pittsburgh
[2] Department of Mechanical Engineering and Materials Science, University of Pittsburgh
Contacts (*): Paolo Zunino, Phone: 412-624-9774, Email: paz13@pitt.edu
Position Papers: DOE Applied Mathematics Research for Exascale Computing

Background and motivations. The forthcoming exascale computing infrastructure will open new possibilities to scientific computing. Problems that are not directly solvable today, because they require excessive computational efforts and are thus addressed using reduced modeling techniques, will soon fall in the range of direct simulations using exascale computers. We are particularly interested to consider the numerical approximation of partial differential equations on domains with complex shape. In the range of the Department of Energy mission, a typical example is the computational analysis of flows through fractured media, involving irregular and extended fracture systems. Another application is the study of materials with inclusions characterized by heterogeneous physical properties or shapes. The present computational approaches to address these problems are based on homogenization techniques or multiscale Galerkin projections. The former family of methods has proved to be very effective when a clear scale separation is observed between the macroscopic resolution scale and the microscale where fractures or inclusions are observed, see [6, 5] or [2, 7, 1] for a selection of recent works. A secondary assumption requires that the microscale features a regular and periodic pattern. These conditions are not always verified in realistic applications, where the interplay of multiple and overlapping spatial scales may dominate the behavior of the system. The latter methods, namely multiscale or multiresolution techniques, are very successful in addressing these issues [10, 3, 8, 9]. However, when applied to the exascale computing framework, these methods may feature limitations in achieving extreme levels of concurrency. This is due to the fact that these techniques are based on the interaction between coarse grid solvers, usually defined by means of multiscale basis functions, and local solvers aiming to capture the shape of problem dependent basis functions on local sub-domains. An expert application of load balancing techniques is usually required to efficiently implement such schemes on high performance computing platforms. We believe that exascale computers will enable scientists to override these drawbacks by simply resorting to direct simulations of the problems at hand. However, as discussed below, the interplay between

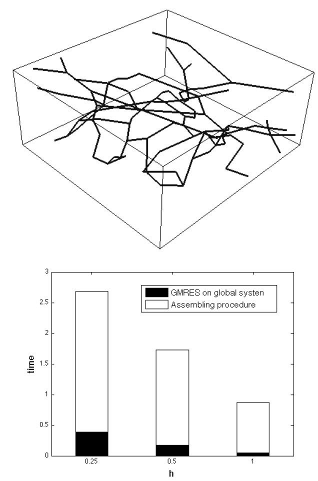

Figure 1: (top) The geometrical model. (bottom) CPU time devoted to searching and building up interpolation matrices between the network and the bulk volume (white columns) compared to solution time (black columns) for decreasing values of the mesh size.

numerical solvers and computational geometry will assume a paramount importance in the design and implementation of this new class of physical simulators.

Algorithms. In our view, the most appropriate computational framework to address problems with a large number of inclusions is the family of Partition of Unity (PUM) or eXtended Finite Element Methods (XFEM) [14, 13, 4, 11]. Two distinctive features of these methods are the following: (i) the ability to build the approximation on meshes which may be partially independent of the geometry of the problem domain; (ii) the ability of incorporating any special function of interest into the construction of the approximation. We are particularly interested in the new computational capabilities introduced by the former property, which allows to shift the definition of geometrical details of the problem from the mesh to the space of problem parameters. In other words, the topology of the inclusions now becomes an independent entity that complements the dataset of problem parameters. This new setting facilitates the treatment of extremely complex configurations, but opens new unexplored issues at the level of computational geometry. More precisely, the implementation of PUM/XFEM solvers for problems with complex configurations of inclusions requires to determine and organize the intersection of two geometric structures. One is the computational mesh relative to the finite element solver, which may be in this case a quasi-uniform partition of the whole domain where the problem is defined. The other is a large data structure describing the topology of the inclusions. The application of efficient search algorithms to map the inclusions onto the computational mesh, as well as the definition of appropriate data structures, become fundamental aspects of the problem, in order to maximize the locality of discrete algebraic operators and minimize the communication between processors. Figure 1 shows the issues arising if these aspects are not addressed efficiently, when a scalar FEM solver is applied to analyze the interaction between a homogeneous porous medium and a network of one-dimensional channels. At the algebraic level, the introduction of the additional degrees of freedom characteristic of PUM/XFEM gives rise to large sparse systems with block structure, where each block accounts for one individual entity in the dataset of inclusions. High level of concurrency could be achieved by simplifying or cutting off the long range interactions between inclusions, because the convex hull of the global matrix would be progressively reduced. In this respect, we see some analogies of this problem with the quantification of forces between a many body system. We believe that ideas coming from the fast multipole method, [12], can help with designing algorithms to account for large numbers of inclusions.

Software. At the level of available software libraries, the computational mechanics community is aware of a substantial gap in the integration of computational geometry with simulation tools. As an example, isogoemetric analysis is rapidly proceeding towards the unification of computer aided design and finite element analysis. We believe that other directions of research are equivalently interesting, but still unexplored. We propose here to strengthen the interaction of algorithms devoted to searching and interpolation with finite element open source packages. Ongoing projects that are proceeding in this directions, such ad Getfem++ (`download.gna.org/getfem/html/homepage/`) and Fenics (`fenicsproject.org/`) are increasing their popularity among users. In our opinion, the most complete computational geometry library currently available to the public is the Computational Geometry Algorithm Library, CGAL, (`www.cgal.org/`). Still many steps have to be taken to achieve a full and efficient integration of these tools into finite element solvers.

References

[1] T. Arbogast. Homogenization-based mixed multiscale finite elements for problems with anisotropy. *Multiscale Modeling and Simulation*, 9(2):624–653, 2011.

[2] T. Arbogast and H.L. Lehr. Homogenization of a darcy-stokes system modeling vuggy porous media. *Computational Geosciences*, 10(3):291–302, 2006.

[3] T. Arbogast, G. Pencheva, M.F. Wheeler, and I. Yotov. A multiscale mortar mixed finite element method. *Multiscale Modeling and Simulation*, 6(1):319–346, 2007.

[4] I. Babuška and J. M. Melenk. The partition of unity method. *Internat. J. Numer. Methods Engrg.*, 40(4):727–758, 1997.

[5] Ivo Babuška. Homogenization approach in engineering. In *Computing methods in applied sciences and engineering (Second Internat. Sympos., Versailles, 1975), Part 1*, pages 137–153. Lecture Notes in Econom. and Math. Systems, Vol. 134. Springer, Berlin, 1976.

[6] J. Bear. *Dynamics of Fluids in Porous Media*. Dover books on physics and chemistry. Dover Publications, Incorporated, 2013.

[7] J. Douglas Jr., T. Arbogast, P.J. Paes-Leme, J.L. Hensley, and N.P. Nunes. Immiscible displacement in vertically fractured reservoirs. *Transport in Porous Media*, 12(1):73–106, 1993.

[8] Y. Efendiev, J. Galvis, and E. Gildin. Local-global multiscale model reduction for flows in high-contrast heterogeneous media. *Journal of Computational Physics*, 231(24):8100–8113, 2012.

[9] Y. Efendiev, J. Galvis, and X.-H. Wu. Multiscale finite element methods for high-contrast problems using local spectral basis functions. *Journal of Computational Physics*, 230(4):937–955, 2011.

[10] Yalchin Efendiev and Thomas Y. Hou. *Multiscale finite element methods*, volume 4 of *Surveys and Tutorials in the Applied Mathematical Sciences*. Springer, New York, 2009. Theory and applications.

[11] T.-P. Fries and T. Belytschko. The extended/generalized finite element method: An overview of the method and its applications. *International Journal for Numerical Methods in Engineering*, 84(3):253–304, 2010.

[12] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *Journal of Computational Physics*, 73(2):325–348, 1987.

[13] J. M. Melenk and I. Babuška. The partition of unity finite element method: basic theory and applications. *Comput. Methods Appl. Mech. Engrg.*, 139(1-4):289–314, 1996.

[14] T. Strouboulis, K. Copps, and I. Babuška. The generalized finite element method. *Comput. Methods Appl. Mech. Engrg.*, 190(32-33):4081–4193, 2001.